



Технології розподілених обчислень

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Другий (магістерський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>123 Комп'ютерна інженерія</i>
Освітня програма	<i>Комп'ютерні системи та мережі</i>
Статус дисципліни	<i>Вибіркова</i>
Форма навчання	<i>Очна (денна)</i>
Рік підготовки, семестр	<i>1 курс, весняний семестр</i>
Обсяг дисципліни	<i>4 кредитів ЄКТС</i>
Семестровий контроль/ контрольні заходи	<i>Екзамен</i>
Розклад занять	<i>Лекцій 14 (28 годин), Лабораторні роботи -18 годин, Самостійна робота студентів – 99 годин, МКР</i>
Мова викладання	<i>Українська/Англійська</i>
Інформація про керівника курсу / викладачів	<i>Лектор: асистент кафедри обчислювальної техніки, Кочура Юрій Петрович, iuriy.kochura@gmail.com Лабораторні: асистент кафедри обчислювальної техніки, Кочура Юрій Петрович, iuriy.kochura@gmail.com</i>
Розміщення курсу	<i>https://courses-cs-kpi.github.io/pd-22sp/</i>

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

У рамках цього курсу ми будемо вивчати як програмувати паралельні та розподілені обчислювальні системи та створювати програми, які б володіли:

- Високою продуктивністю та енергоефективністю;
- Функціональністю та масштабованістю.

Навчальна мета

- З'ясувати основні відмінності між латентними пристроями (ядра CPU) та пропускними пристроями (ядра GPU);
- Зрозуміти, чому найкращі програми-переможці все частіше використовують пристрої обох типів.

Дисципліна "Технології розподілених обчислень" забезпечує наступні програмні компетентності і програмні результати освітньо-наукової програми (ОНП): ФК11, ПРН6.

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Для проходження цього курсу потрібно володіти наступними навичками:

1. Базовий рівень володіння англійською мовою не нижче А2.
2. Базові знання C/C++ та Python: типи даних, цикли, умови, функції та маніпуляції з масивом.

3. Обізнаність з NumPy, включаючи використання ndarrays та ufuncs.

3. Зміст навчальної дисципліни

1. Вступ до гетерогенного паралельного програмування
2. Вступ до CUDA C
3. Модель паралелізму CUDA
4. Модель пам'яті та локальність
5. Ефективність виконання потоку
6. Продуктивність доступу до пам'яті
7. Шаблони паралельних обчислень
8. GPU як частина архітектури ПК
9. Ефективна передача даних хост-пристрій
10. Модель програмування MPI
11. CUDA Python з використанням Numba
12. Запуск ядра на кількох графічних процесорах
13. Прискорення додатків за допомогою конкурентних потоків

4. Навчальні матеріали та ресурси

Базова:

1. David Kirk and Wen-mei W. Hwu (2017). *Programming massively parallel processors: a hands-on approach*, 3rd Edition.
2. Robert (Bob) Robey and Yuliana (Yulie) Zamora (2021). *Parallel and High Performance Computing*.
3. Технології розподілених обчислень. Інструкції для виконання лабораторних робіт. Навч. посібник для здобувачів ступеня магістр за спеціальністю 123 «Комп'ютерні системи та мережі» / Гордієнко Ю. – К.: КПІ ім. Ігоря Сікорського, 2022. – 20с. [Електронний ресурс], <http://comsys.kpi.ua>

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

5.1. Лекційні заняття

№ лекції	Назва лекції та перелік основних питань	Кількість ауд. годин
1	<i>Вступ до гетерогенного паралельного програмування</i>	2
2	<i>Вступ до CUDA C</i>	4
3	<i>Модель паралелізму CUDA</i>	2

4	<i>Модель пам'яті та локальність</i>	2
5	<i>Ефективність виконання потоку</i>	2
6	<i>Продуктивність доступу до пам'яті</i>	2
7	<i>Шаблони паралельних обчислень</i>	2
8	<i>GPU як частина архітектури ПК</i>	2
9	<i>Ефективна передача даних хост-пристрій</i>	2
10	<i>Модель програмування MPI</i>	2
11	<i>CUDA Python з використанням Numba</i>	2
12	<i>Запуск ядра на кількох графічних процесорах</i>	2
13	<i>Прискорення додатків за допомогою конкурентних потоків</i>	2
	Разом:	28

5.2. Лабораторні/практичні роботи

№ з/п	Назва лабораторної/практичної роботи	Кількість ауд. годин
1	<i>Основи прискорених обчислень з CUDA C/C++</i>	6
2	<i>Основи прискорених обчислень з CUDA Python</i>	4
3	<i>Масштабування робочих навантажень на кількох графічних процесорах</i>	4
4	<i>Прискорення додатків за допомогою конкурентних потоків</i>	4
	Разом:	18

6. Самостійна робота студента/аспіранта

- підготовка до лекцій
- підготовка до практичних занять
- підготовка до заліку

7. Політика навчальної дисципліни (освітнього компонента)

Ви можете обговорювати завдання практичних робіт у групах. Однак, кожен студент/студентка повинен/повинна підготувати розв'язки завдань самостійно. Під час проходження цього курсу Ви зобов'язані дотримуватись Кодекс честі КПІ ім. Ігоря Сікорського та усі наступні правила:

1. Кожен з Вас повинен відправляти на перевірку власно виконану роботу. Використання чужих розв'язків або програмного коду і представлення їх за свої напрацювання є плагіатом та серйозним порушенням основних академічних стандартів.

2. Ви не повинні ділитися своїми розв'язками з іншими студентами, а також просити інших ділитися своїми розв'язками з Вами.

3. Якщо Ви отримували допомогу у вирішенні певного завдання, Ви маєте зазначити це у звіті, а саме: від кого та яку допомогу отримали.

8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Поточний контроль: [МКР](#)

Календарний контроль: провадиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу.

Семестровий контроль: [залік](#)

Система оцінювання

40% Практичні завдання (10% кожне)

10% Контрольна

20% Семінар

30% Залік

Важливо! Умова допуску до семестрового контролю (заліку):

Практичні завдання + Проект \geq 42%

Шкала оцінок КПІ ім. Ігоря Сікорського:

A = 95–100	Відмінно
B = 85–94	Дуже добре
C = 75–84	Добре
D = 65–74	Задовільно
E = 60–64	Достатньо
F < 60	Незадовільно
Fx < 42	Недопущений
Порушення кодексу честі	Усунений

9. Додаткова інформація з дисципліни (освітнього компонента)

- можливість зарахування сертифікатів проходження дистанційних чи онлайн курсів за відповідною тематикою;
-

Робочу програму навчальної дисципліни (силабус):

Склав асистент кафедри обчислювальної техніки, Кочура Ю. П.

Ухвалено кафедрою обчислювальної техніки (протокол № 10 від 25.05.2022)

Погоджено Методичною комісією факультету інформатики та обчислювальної техніки (протокол № 10 від 09.06.2022)